

Application No. 09/705,766

AMENDMENTS TO THE CLAIMS

A detailed listing of all claims that are, or were, in the present application, irrespective of whether the claim(s) remains under examination in the application are presented below. The claims are presented in ascending order and each includes one status identifier. Those claims not cancelled or withdrawn but amended by the current amendment utilize the following notations for amendment: 1. deleted matter is shown by strikethrough for six or more characters and double brackets for five or less characters; and 2. added matter is shown by underlining.

1. (Cancelled) A method for distributing incoming client requests across multiple servers in a networked client-server computer environment, said method comprising the steps of:
 - (a) collecting at least two client requests incoming within a predetermined time interval;
 - (b) analyzing each said collected client requests with respect to at least one attribute;
 - (c) collecting resource capability information of each server;
 - (d) upon completion of said time interval, distributing said collected client requests across the multiple servers in response to in response to the attributes of said collected client requests and the resource capability information of the multiple servers; and
 - (e) repeating steps (a) through (d) for subsequent ones of said time interval.
2. (Cancelled) The method of Claim 1, wherein the attributes analyzed in step (b) are selected from the set comprising: categorical criteria and demographic criteria
3. (Amended) The method of Claim ~~[[1]]~~23, wherein said resource capability information comprises a resource capability metric for each of at least five resource parameters for each

Application No. 09/75,766

server, regarding the server's CPU availability, memory availability, connectivity to storage, connectivity to a proxy server, and connectivity to peer servers.

4. (Amended) The method of Claim 3, wherein said method ~~wherein step (a)~~ further comprises the step of providing a dynamic, relational database and process of statistical inference for ascertaining expected demand patterns involving said at least one attribute.

5. (Original) The method of Claim 4, wherein the number of said expected demand patterns can dynamically increase or decrease.

6. (Original) The method of Claim 4, wherein a resource requirement metric for each of said at least five resource parameters is assigned to each said expected demand pattern, wherein each said collected request is identified with at least one said expected demand pattern, and wherein said resource requirement metrics assigned to said identifying expected demand pattern are further assigned to said collected request.

7. (Amended) The method of Claim 6, ~~wherein the step (d)~~ further ~~comprises~~ comprising ~~the steps of~~ determining the metric distance between said resource requirement metrics and said resource capability metrics for at least one combination of said collected client request and server pairings and selecting a server for each said collected request so that the sum of said metric distances for said at least one combination of said pairings is minimized.

8. (Amended) The method of Claim 7, wherein an optimization paradigm is used to at least partially perform ~~step (d)~~ the step of selecting a server for each said collected request so that the sum of said metric distances for said at least one combination of said pairings is minimized

9. (Canceled) A method for distributing incoming client requests across multiple servers in a networked client-server computer environment, said method comprising the steps of:

- (a) collecting at least two requests incoming within a predetermined time interval;

Application No. 09/715,766

- (b) analyzing each of said collected request with respect to at least one attribute;
- (c) analyzing said at least one attribute for ascertaining statistical pattern across said collected requests;
- (d) identifying at least one resource parameter for said servers.
- (e) collecting a resource capability metric of each server for said at least one resource parameter;
- (f) assigning a resource need metric for said at least one resource parameter to each statistical pattern;
- (g) correlating each of said collected requests with at least one said statistical pattern;
- (h) assigning said metric for said at least one resource parameter assigned to said statistical pattern correlated to said collected request;
- (i) determining a metric distance between a resource need metric and a resource capability metric for at least one combination of a pairing of said collected request and one of said servers;
- (j) upon completion of said time interval, selecting a server for each of said collected request so that a sum of said metric distances for said pairings is minimized; and
- (k) repeating steps (a) through (j) for consecutive ones of said time intervals.

10. (Canceled) The method of Claim 9, wherein the attributes analyzed in step (b) are selected from the set comprising: categorical criteria and demographic criteria.

11. (Canceled) The method of Claim 9, wherein said at least one resource capability metric comprises metrics chosen from the set comprising: the server's CPU availability, memory availability, connectivity to storage, connectivity to a proxy server, connectivity to peer servers, and connectivity to the Internet.

Application No. 09/715,766

12. (Canceled) The method of Claim 11, wherein step (g) said method further comprises the step of providing a dynamic, relational database and process of statistical inference for ascertaining said statistical patterns.
13. (Canceled) The method of Claim 9, wherein the number of said statistical patterns can dynamically increase or decrease.
14. (Canceled) The method of Claim 9, wherein an optimization paradigm is used to select a server.
15. (Canceled) A system for distributing incoming client requests across multiple servers in a networked client-server computer environment, said system comprising;
- a request table to collect at least two requests incoming within a predetermined time interval;
 - a request examiner routine to analyze each said collected request with respect to at least one attribute;
 - a system status monitor to collect resource capability information of each server;
 - and
 - an optimization and allocation process to distribute said collected requests in said table across the multiple servers upon completion of said time interval in response to said request table and said resource capability information.
16. (Canceled) The system of Claim 15, wherein there are at least two said time intervals and said time intervals are consecutive.
17. (Canceled) The system of Claim 15, wherein the table is maintained as a relational database and further comprising process of statistical inference to ascertain expected demand patterns involving said at least one attribute.

Application No. 09/75,766

18. (Amended) The system of Claim [[15]] 24, wherein said resource capability information comprises a resource capability metric for each of at least a plurality of resource parameters for each server selected from the set comprising: the server's CPU availability, memory availability, connectivity to storage, connectivity to a proxy server, connectivity to peer servers, and connectivity to the Internet.

19. (Amended) The system of Claim [[15]] 24, wherein the relational database is dynamic and wherein the number of said expected demand patterns can dynamically increase or decrease.

20. (Amended) The system of Claim [[15]] 24, wherein a resource requirement metric for each of said at least five resource parameters is assigned to each of said expected demand pattern, wherein each said collected request is correlated with at least one said expected demand pattern, and wherein said resource requirement metrics assigned to said correlated expected demand pattern are further assigned to said collected request.

21. (Original) The system of Claim 20, wherein the process of distributing said collected requests further comprises determining the metric distance between said resource requirement metrics and said resource capability metrics for at least one combination of a pairing of said collected request and server and selecting a server for each said collected requests so that the sum of said metric distances for said at least one combination of said pairings is minimized.

22. (Original) The system of Claim 21, wherein a global optimization paradigm is used to minimize a total assignment cost for said pairings of all of said servers and said requests.

23. (Newly Presented) A method for distributing incoming client requests across multiple servers in a networked client-server computer environment comprising:

(a) collecting client requests incoming within a predetermined time interval;

Application No. 09/75,766

- (b) upon receipt of the at least two incoming client requests within the predetermined time interval, analyzing each of the client requests using categorical criteria and demographic criteria to extract attributes of the request;
- (c) classifying the client requests based on the extracted attributes by comparing each request with a pattern selected from a set of patterns in an adaptive request table to find a match-pattern that best matches the request;
- (d) using the match-pattern to associate a requirements vector with each request, the requirements vector being populated with at least five resource parameters that describe the expected resource requirements of the request;
-
- (e) capturing resource capability information for each server at least once during the predetermined time interval, each server being associated with a capability vector refreshed with the resource capability information;
- (f) following steps (d) and (e), for each client request and server pair, computing a vector space distance between the requirement vectors and capability vectors corresponding to the client request and the server respectively, the vector space distance being an update to an element in a cost matrix initialized at the start of the predetermined time interval
- (g) at the completion of the time interval, distributing the client requests across multiple servers to minimize a cost metric associated with the cost matrix for all combinations of client requests and server resource capabilities; and
- (h) repeating steps (a) through (g) for subsequent ones of said time intervals after initializing the cost matrix.

Application No. 09/7: 5,766

24. (Newly Presented) A system for servicing multiple requests to be distributed across multiple servers in a networked client-server computer environment, the system comprising:

a request examiner routine for analyzing each of the client requests using categorical criteria and demographic criteria to extract attributes of the request;

a request table for collecting attribute information associated with each client request incoming within the predetermined time interval;

an adaptive request table populated with a set of patterns, each pattern associated with a generic request type that is most likely to be received by a proxy server, the adaptive request table suited for classifying client requests by comparing each request with a pattern selected from the set of patterns in the adaptive request table to find an expected demand pattern that best matches the client request resulting in a successful match, so that an expected demand on resources by the client request can be predicted using the expected demand pattern in the adaptive request table;

a relational database coupled with a process of statistical inference to facilitate the construction of an adaptive request table, the adaptive request table being updated by the process of statistical inference for each successful match of the client request with the expected demand pattern from the set of patterns;

a resource table for collecting resource capability information about each server at least once during the predetermined time interval; and

an optimization and allocation process for selecting a server resource for receiving each client request so that a sum of all costs in a cost matrix is minimized for all combinations of client requests and server resources.